Int. J. on Recent Trends in Engineering and Technology, Vol. 10, No. 1, Jan 2014

Energy Efficient Mobile Targets Classification and Tracking in WSNs based on Parameter Priority

Pinaki Sankar Chatterjee, Monideepa Roy
School of Computer Engineering
KIIT University, Bhubaneswar, Odisha
{pinaki.sankar.chatterjee, monideepa.roy}@gmail.com

***Abstract*— Most energy management strategies in WSNs assume that data acquisition takes more time than data transmission. But while testing several applications it has been shown that the sensing activity of some sensors consumes significantly more energy than the radio. Therefore it is very important for a sensor to decide whether an object is its desired target or not before it actually starts sensing its data. The idea here is to altogether avoid sensing the data of an object which is not considered our target. For this it is very necessary to segregate the parameters which will uniquely identify a particular class of objects. At present there are no lightweight object classification algorithms suitable for sensor networks that perform such a filtering at the preliminary stage. Also the main existing approaches for efficient energy management strategies in power-hungry sensors fall under two major categories: duty cycling and adaptive sensing. In this paper, an energy saving mobile target tracking scheme is presented, based on the ID3 algorithm [6] [12] which performs such classification and elimination of undesired objects, to reduce the volume of data acquisitions and transmissions by a sensor.**

***Index Terms*— WSN, ID3 algorithm, mobile target tracking, target classification.**

I. INTRODUCTION

Sensor nodes are used widely for sensing physical environment information, processing the sensed data locally at unit and cluster level, and sending it finally to sinks or base stations. Since most sensor nodes are battery-powered, they have limited lifetime and in a lot of cases not replaceable or rechargeable due to environmental constraints [1] [2] [3]. Therefore energy should be managed judiciously both at sensor node level and cluster node level to prolong the network lifetime as long as possible.

Most of the existing energy management strategies assume that the acquisition of data and its processing consumes significantly less energy than their transmission [4]. So reducing the radio activity will in turn increase the network lifetime [9] [10]. But this is not necessarily true for a several applications where the power consumption of the sensing activity is almost as much as or even more than that of the radio [5]. So an effective energy management strategy should also optimize the use of energy-hungry sensors, since they may also become a major components affecting the network lifetime. One way to do that would be to reduce the number of data acquisitions necessary.

This could be done if we collect the data *only* from those objects which are the targets. For this we need to design and implement a classification mechanism which will filter out the undesired objects and only measure the reading of those objects which are the desired targets, through the sensors.

In the FOMS algorithm proposed in this paper we have used the *ID3 algorithm* [6] [8] [12] to initially

DOI: 01.IJRTET.10.1.1387

classify whether an object is the desired target or not by only checking some important parameters at first. If the object does not conform to those important parameters then it is not tested for the remaining parameters. So no unnecessary power is being used for sensing the data from objects that are not the desired targets. This is helpful in saving battery power of the sensors and thereby increasing the network lifetime. The order of the parameters to be checked is done by calculating their gains then arranging them in the descending order of their gains. Here the sensors are arranged in clusters and each cluster has a cluster head. The object is first checked for the parameter with the maximum gain first, by the cluster head sensor. If the object is a desired target, only then the rest of the sensing parameters located on the leaf nodes are activated and measured. Otherwise the leaf sensors which measure the parameters with lesser gains go back to idle mode. So here the number of parameters sensed as well as number of radio transmissions are also reduced to a considerable extent by reducing the number of measurements done for false positive cases. Also a round robin scheduling process is incorporated here, which is to be done periodically for choosing a cluster head among the various devices in a cluster based on their remaining battery lives so that the energy depletion happens in a more balanced manner.

The rest of the paper is organized as follows: Section 2 discusses the related work, Section 3 explains the ID3 selection algorithm which is the basis for our proposed system, Section 4 presents the proposed system, Section 5 gives the details of the proposed Filtered Object Monitoring and Sensing Algorithm (FOMS), Section 6 is an analysis of the proposed algorithm and finally Section 7 presents the conclusion and future work.

II. Related Work

In this section a brief review of the existing approaches that have been proposed for better energy management [13][14][15] while sensing an object in sensor networks is presented, after which we present the advantage of our proposed algorithm.

Most of the existing monitoring applications for energy saving in sensor networks can be broadly classified under two categories: *duty cycling* and *adaptive sensing* [7].

Duty cycling [16] [17] sensing is useful when the rate of data is time invariant and the dynamics of the phenomenon are known in advance. Here the sensors only wake up during the time needed for acquiring the data to be sensed and are powered off immediately after that. However this method is not very suitable if it is not known in advance, the rate of data acquisition to be made [7]. In such cases, adaptive sensing is used, where the sensor activity dynamically adapts to the actual dynamics of the process being monitored.

Adaptive cycling is further subdivided into hierarchical sensing; adaptive sampling and model based active sampling [5].

In hierarchical sensing, there are different sensors of different accuracy levels. At first the sensing is done at a coarser level, after which the sensing is done at a finer level by the sensors which have more accuracy and thereby are more energy-hungry. In the adaptive sampling methods the sampling rate may be dynamically changed based on the available energy or on the rate of change of generation of the sensed data [18][19][20].

A third technique is model based active sampling where a model of the sensed phenomenon is built on top of the initially sampled data set. Once the model is made, data for some next duration is predicted by the model, instead of acquiring new data through sensors. At the point when the level of accuracy reduces or the model needs to be updated, the sensors are again powered on [5].

In all the above models, sensing is done on all the parameters and on all the objects in the range of the sensors. So it may happen that we are using up crucial sensor battery in tracking objects that are not the desired targets.

In case of multiple-parameter sensing [21][22], it may sometimes be possible to classify whether an object is our desired target or not based on some key parameters, instead of having to measure all the parameters.

However if a selective sensing strategy could be devised, where any new object is first classified whether it is a target or not and only then the sensors start sensing the data then a lot of energy could be saved. We also apply an hierarchical classification scheme, so that here a preliminary classification is done based on a few key parameters first and then tested for other secondary parameters. So if it can be determined from the major parameters that the object is not a target, then battery is not spent on sensing the other parameters of the object. Therefore the vital point here is to isolate the key parameters based on their priority which defines a particular object as a desired target or not.

In this paper we present a method for using the above concept. Here an energy management scheme based on the ID3 classification algorithm is presented which identifies the order of priority of the characteristics which

accurately define an object based on the calculation of the entropy gains of the various parameters of that object. It will then only measure secondary parameters if the object is found to conform to the primary parameters.

III. The ID3 Classification Algorithm

Since the ID3 algorithm forms the basis of the classification algorithm proposed in this paper, so before giving a description of the proposed framework, an overview of the ID3 classification algorithm [6] [8] [12] is given in this section.

The ID3 classification algorithm builds a decision tree from a fixed set of examples. The resulting tree that is formed is then used to classify future samples. The example typically has several attributes and belongs to a class (like yes or no). The leaf nodes of the decision tree contain the class name whereas a non-leaf node is a decision node. The decision node is an attribute test with each branch (to another decision tree) being a possible value of the attribute. ID3 searches through the attributes from the examples and aims to extract the attribute that best separates the given examples. This algorithm uses a greedy search, i.e. it picks the best attribute and does not backtrack to reconsider the choices made earlier.

To decide which attribute is the best, ID3 uses a statistical property called information gain. Gain measures how well a given attribute separates training examples into targeted classes. The one with the highest information (information being the most useful for classification) is selected. To define gain, a concept from information theory called entropy is used. Entropy measures the amount of information in an attribute.

Given a collection **S** of **c** outcomes, then:

$$Entropy\,(S) = \sum -p(I) \times \log_2 p(I)$$

where p(I) is the proportion of S in the class I, and $\sum$ is over c.

For example, if S is a collection of 4 samples with 3 YES and 1 NO, then:

$$Entropy(S) = -(3/4)\log_2(3/4) - (1/4)\log_2(1/4) = 0.81127$$

If the Entropy is zero it means “perfectly classified” and if it is one then it means “totally random”.

Information Gain calculates the effective change in entropy after making a decision based on the value of an attribute. Now Gain(S, A) is information gain of example set S on attribute A is defined as:

$$Gain(S, A) = Entropy(S) - S((|S_y| / |S|) \times Entropy(S_y))$$

Here, S_v = Subset of S for which attribute A has value v.

$|S_v|$= Number of elements in S_v.

|S|=Number of elements in S.

IV. The Proposed System

The objective of the proposed system is to track the entry of only the desired categories of objects in the area of surveillance, which is henceforth referred to as ‘targets’, in the rest of this paper. The classification of whether the new object of entry is the desired target or not is done based on some parameters, which are target-specific and may vary according to the type of application the algorithm is being used for.

Here we consider a test area which is covered by clusters of sensors, with each cluster having a cluster head and a number of leaf nodes, as shown in Figure-3. There is provision for measuring N different types of parameters of an object, on each cluster of sensor nodes, e.g. height measurement, speed measurement, temperature measurement, and sound/vibration measurement. The cluster head only senses the parameter with the highest gain and has a higher sensing range than the leaf nodes, while the leaf sensors in the cluster are multiparameter sensors and measure the rest of the parameters. Based on the sensing output of these sensors it can be classified whether an object is the desired target or not. In this paper, for generality the parameters to be sensed are denoted as parameters x, y, z and k.

If all the x, y, z and k parameters of each object are measured for all new objects, then a lot of power is wasted for the acquisition and transmission of data for objects which may not be our desired targets, and the battery power of the sensor nodes will drain out very fast. The battery power for the sensor nodes can be conserved and the lifetime of the WSN may be prolonged if we can somehow reduce the number of data

acquisitions as well as the number of data transmissions.
Here energy optimized Filtered Object Monitoring and Sensing Algorithm is proposed, for this purpose.
Initially the cluster heads of each cluster are in idle mode and all the other nodes are in sleep mode. Whenever a new object enters the test area, the cluster head which first senses the object measures the value of the parameter which is at the root of the decision tree, say x. To determine which parameter will be in the root node of the decision tree, we need to find the parameter with the maximum gain. The maximum gain among x, y, z and k parameters is calculated using the formula in the ID3 classification algorithm as mentioned in the earlier section.
The gains for the rest of the parameters, y, z, and k can be calculated in the same way. The rest of the parameters are measured by the child sensor nodes.
Based on this value of the parameter recorded by the cluster head, if the decision of the sensor is 'NO' then it is decided that the object is not the intended object. The cluster head then goes back to its previous idle state without doing any further processing for this object. If the decision is YES then the cluster head activates the leaf sensors that in turn sense the object for the remaining parameters, based on the decreasing order of gain.
For example to classify if the object is human or not we may take the height measurement sensor as in Figure 1to have the maximum gain and it has three types of outputs.

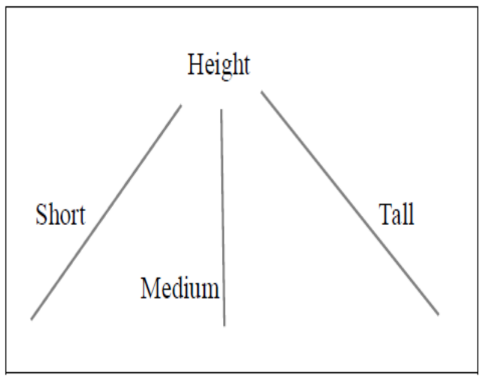

Figure 1 –Sensor with the highest gain

If the height is 'Short' (threshold value between 1 to 3.11 feet) or 'Tall' (threshold value between 8.1 to 20 feet) then from this it can be concluded that the object is definitely not human. If the height is 'Medium' (threshold value between 4 to 8 feet), then only it is passed on for testing with the speed measurement sensor as speed has got the next highest gain. Again the speed parameter has got three values like 'Slow' (threshold value between 0 to 10 kilometer/hour), 'Moderate' (threshold value between 11 to 30 kilometer/hour) and 'Fast' (threshold value between 31 to 200 kilometer/hour) shown in Figure 2.
If the output is Slow then only it is passed for the next level testing to confirm the object is human or not. If the output is 'Moderate' or 'Fast' then the object may be a small car or medium height animal running in moderate speed which are not of our intended object and they are not tracked.
The entire process is repeated whenever a new object enters the test area.
Additionally, to ensure a more uniform depletion of energy among the sensor nodes in the cluster, a cluster head election algorithm is also used here. The system periodically measures the remaining battery life for all the sensor nodes in a cluster and whenever the remaining energy of a cluster head falls below the level of any one of its leaf nodes, then the cluster head election algorithm is run to elect the leaf node which has the most battery life, as the new cluster head.

V. The Filtered Object Monitoring And Sensing Algorithm (FOMS)

The first part of the algorithm checks whether a new object entering the area is the desired target or not. For this the cluster heads are set to idle mode and the leaf nodes are set to sleep mode. Whenever a new object enters the test area, the cluster head becomes active and the following steps are repeated. The cluster head checks the parameter with the highest gain and if the object is the desired target then it will activate the leaf node sensors for measuring the next parameter.

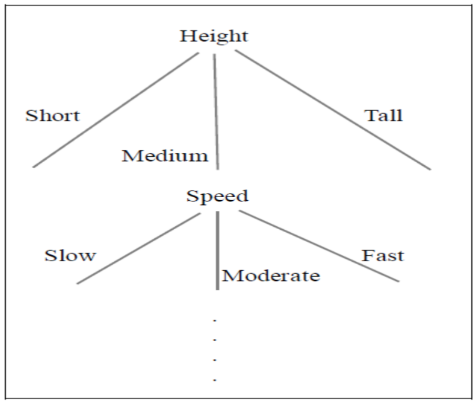

Figure 2- Sensor with next highest gain

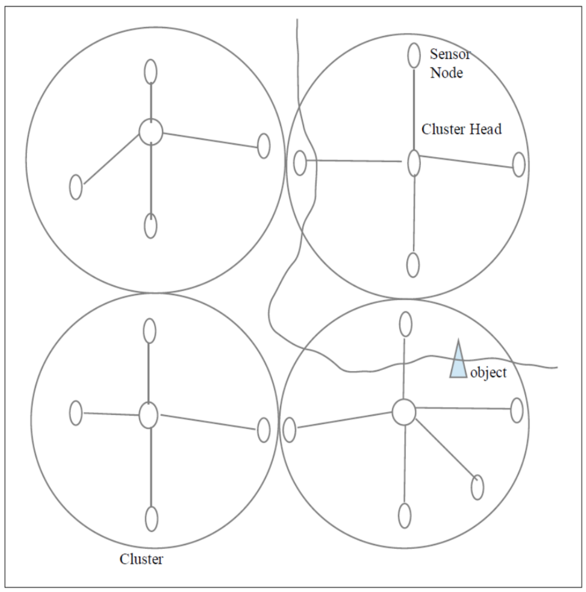

Figure 3 – The WSN topology with clusters monitored by cluster heads

Algorithm FOMS( )
// Let X, Y, Z and K be the parameters to be sensed.
// Cluster head senses the parameter with maximum gain, say X.

1. ***If*** *node_type* **==** *cluster_head,* ***then*** *set status = "idle"*
   elseif *node_type == leaf_node* ***then*** *set status = "sleep"*
2. ***If*** *new_ object enters in test area* ***then repeat*** *steps 3 through 13*
3. ***Begin***
4. *Set cluster_head status = "active"*
5. ***Read*** *X*
6. ***If*** *X_value == "false"* ***then*** *new_object = "not a target"* ***And*** *Set cluster_head status = "idle"*
7. ***Elseif*** *X_value = "true"* ***then***
8. ***Begin***
9. *Set leaf_node status = "active"*
10. ***Read*** *Y*
11. ***End***
12. ***End***

The Cluster Head Selection Algorithm (CHS)

Since the cluster head is needed to be in active or idle mode all the time, therefore it has a faster depletion of energy than the other leaf nodes.

Algorithm CHS ( )

1. ***repeat***
2. ***for*** *each cluster 1* ***to*** *n*
3. ***begin***
4. ***if (****battery_life(cluster_head) <= battery_life(leaf_node))*
5. ***then*** *set node with maximum battery_life = new_cluster_head*
6. ***end***
7. ***Until (****battery_life <= min_threshold_ value)*

To make this energy depletion more uniform and increase the network lifetime, the job of functioning as a cluster head is periodically shuffled between the leaf nodes in a cluster in a round robin manner and the next leader is the leaf node with the highest remaining battery life at that point in that cluster.

VI. Results And Analysis

We have considered a scenario where our aim is to detect only those vehicles on the street which are traveling above a certain speed threshold and gather relevant data about them. A total of 40 vehicles were tested for our data set. The cluster heads are equipped with various types of sensors including speed sensors. For this experiment the speed parameter is set to highest gain and the cluster heads only activate the speed sensors and measure the speeds of all the vehicles that pass through a particular test zone. If the cluster head finds that a vehicle is travelling above the threshold value only then it will activate the leaf sensors for measuring the parameter with the next highest gain value, say the heat sensor. We measure the amount of energy consumed for the above scenario by two methods. The first way is by our proposed FOMSA method and the second way is with the traditional method of measuring data. The average amount of power consumed by a specific type of sensor is taken to be according to the data in Table -1 [11].

We propose to show the respective energy consumptions for target parameter sensing using the traditional technique mathematically as follows:

Suppose there are p_1, p_2, p_3, … , p_m parameters for testing each object and the energy consumed by each sensor for sensing each of these parameters are e_1, e_2, e_3, …, e_m respectively. So if there are n objects that are to be sensed in an area of coverage, then the total energy consumption at each cluster will be:

$$E_{total} = n \times \sum_{i=1}^{m} e_i \times p_i$$

This will be the total energy consumption, even if there are only k actual targets for which we need to monitor data, out of the total n objects, if the existing sensing techniques are used.

However, using the proposed classification algorithm, all parameters are not monitored or sensed for all the objects. At first only the parameter with the highest gain is monitored by the cluster head, which decides whether the object is the desired target or not. If the object entering into the coverage area is our intended category of target, then only, the rest of its parameters are sensed. If the object is not our intended target, then the energy consumption is limited to the sensing of the parameter with the highest gain only and not the other parameters, thereby saving the power consumption.

So in this case the total energy consumption per cluster will be:

$$E_{FOMSA} = (n \times e_1 \times p_1) + (k \times \sum_{i=2}^{m} e_i \times p_i)$$

So the energy savings in our approach is $E_{total} - E_{FOMSA}$

$$=(n\times\sum_{i=1}^{m}(e_i\times p_i))-((n\times e_1\times p_1)-k\times\sum_{i=2}^{m}(e_i\times p_i))$$

$$=(n\times e_1\times p_1)+n\times\sum_{i=2}^{m}(e_i\times p_i)-(n\times e_1\times p_1)-k\times\sum_{i=2}^{m}(e_i\times p_i)\ =(n-k)\times\sum_{i=2}^{m}(e_i\times p_i)$$

Battery power spending is for the highest gain parameter sensor for all the cases. However, if k objects are the actual targets out of n objects then for unintended objects battery power can be saved for the parameters sensing second highest gain parameter onwards.
The above two formulas have been applied on our real time local traffic data set to compare the energy consumptions of the two algorithms. The results are shown in Figure-4

TABLE I: POWER CONSUMPTIONS OF DIFFERENT TYPES OF SENSORS

Sensor	Sensing	Power Consumption
Honeywell 1GT	Speed	20 mA
iMEMS	Accelerometer	30 mW
STCN75	Temperature	0.4 mW
MAG3110	Magnetometer	8.6 µA

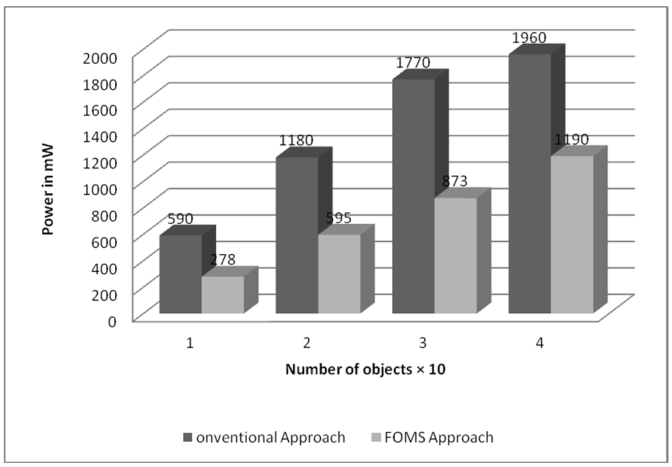

Figure-4: Comparison of energy consumptions per object per cluster

In Figure-4, the black bars represent the power consumption of the sensors in mW by the traditional method where all the sensors are activated at one time while testing each of the 40 vehicles. The gray bars represent the energy consumed by sensors in mW using the FOMSA method where the sensors are activated according to their gain priority. The graphs show that the power consumption of the proposed approach is better than the traditional approach in the network.

VII. CONCLUSION AND FUTURE WORK

In this paper, a scheme for reducing energy consumption in sensors has been presented by preventing the unnecessary tracking of objects which are not our desired targets. It implements hierarchical filtered sensing at the cluster head level, and the priority of the parameters to be sensed is calculated by the ID3 classification algorithm. In this way, the energy that would have been consumed in sensing the objects which are not the desired targets is reduced to a large extent, thereby effectively prolonging the lifetime of the sensor network. Moreover since cluster heads have higher transmission ranges and are in either idle or active modes, the rate of energy dissipation is more on these nodes. So a cluster head selection algorithm has also been implemented which is run periodically to ensure that the energy depletion of the sensor nodes happen at a more uniform rate.
This algorithm may be adapted for various applications where there is a variety of streaming data entering the area of testing and only a particular category of that data is our desired target set for sensing.

In future we plan to extend this algorithm to actually detect various categories of objects rather than merely detecting whether an object is the desired target or not. This type of classification algorithms are especially useful for surveillance or monitoring applications, where a large number objects may enter the test zone but only a few of them are the actual desired targets which need to be monitored.

REFERENCES

[1] I.F.Akyildiz, W. Su, Y. Sankarasubramaniam, E. Capirci, "Wireless Sensor Networks: a Survey", Computer Networks, Vol 38, N.4, March 2002

[2] D. Estrin, L. Girod, G. Pottie, M. Srivastava, "Instrumenting the world with wireless sensor networks", Proc. Acoustics, Speech and Signal Processing, Vol. 4, 2001

[3] K. Romer, F. Mattern, "The design space of wireless sensor networks", Wireless Communications, IEEE, Vol. 11, no.6, Dec. 2004

[4] G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, "Energy Conservation in Wireless Sensor Networks", Ad Hoc Networks, to appear(Currently available at http://info.iet.unipi.it/~anastasi/papers/adhoc08.pdf)

[5] V. Raghunathan, S. Ganeriwal, M. Srivastava, "Emerging Techniques for Long Lived Wireless Sensor Networks", IEEE Communications Magazine, Vol. 44, N. 4, April 2006.

[6] A. Colin, "Building Decision Trees with the ID3 Algorithm", Dr. Dobbs Journal, June 1996.

[7] D. Ganesan, A. Cerpa, W. Ye, Y. Yu, J. Zhao, D. Estrain, "Networking Issues in Wireless Sensor Networks", Journal of Parallel and Distributed Computing, Vol. 64, 2004.

[8] P. E. Utgoff, "Incremental Induction of Decision Trees", Kluwer Academic Publishers, 1989.

[9] R. Want, K. I. Farkas, C. Narayanaswami, "Energy Harvesting and Conservation", IEEE Pervasive Computing, Vol. 4, Issue 1, Jan-Mar. 2005.

[10] C. Alippi, C. Galperti, "An Adaptive System for Optimal Solar Energy Harvesting in Wireless Sensor Network Nodes", Circuits and Systems I: Regular Papers, IEEE Transactions on Vol. 55, 6th July 2008.

[11] C. Alippi, G. Anastasi, M. D. Francesco, M. Roveri, "Energy Management in Wireless Sensor Networks with Energy-hungry Sensors", IEEE Instrumentation and Measurement Magazine, Vol. 12, N.2, April 2009.

[12] http://www.cise.ufl.edu/~ddd/cap6635/Fall-97/Short-papers/2.htm

[13] G. Pottie, W. Kaiser, "Wireless integrated network sensors", Communication of ACM, vol 43, N. 5, 2000, pp. 51–58.

[14] V. Raghunathan, C. Schurghers, S. Park, M. Srivastava, "Energy-aware wireless microsensor networks", IEEE Signal Processing Magazine, 2002, pp. 40–50.

[15] J. Li, P. Mohapatra, "Analytical modeling and mitigation techniques for the energy hole problem in sensor networks", Pervasive Mobile Computing 3, 2007, pp. 233–254.

[16] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, J. Anderson, "Wireless sensor networks for habitat monitoring, in: Proc. ACM Workshop on Wireless Sensor Networks and Applications", Atlanta, USA, September 2002, pp. 88–97.

[17] A. Warrier, S.Park J. Mina, I. Rheea, "How much energy saving does topology control offer for wireless sensor networks? – a practical study", Elsevier/ACM Computer Communications, vol. 30, 2007, pp. 2867–2879.

[18] E. Fasolo, M. Rossi, J. Widmer, M. Zorzi, "In-network aggregation techniques for wireless sensor networks: a survey", IEEE Wireless Communications, vol. 14, 2007, pp. 70–87.

[19] S.S. Pradhan, K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction", IEEE Transactions on Information Theory, vol. 49, 2003, pp. 626–643.

[20] C. Tang, C.S. Raghavendra, "Compression Techniques for Wireless Sensor Networks", Book Wireless Sensor Networks, Kluwer Academic Publishers, 2004, pp. 207-231.

[21] M. Wu, C.W. Chen, "Multiple Bit Stream Image Transmission over Wireless Sensor Networks", Book Sensor Network Operations, IEEE & Wiley Interscience, 2006, pp. 677-687.

[22] Z. Xiong, A.D. Liveris, S. Cheng, "Distributed source coding for sensor networks", IEEE Signal Processing Magazine, vol. 21, 2004, pp. 80–94.